"XANADU: Hypertext from The Future"

by "Roger Gregory" (act. Chip Morningstar)

fm Dr. Dobbs, Jn 83.

XANADU

Hypertext from the Future

by Roger Gregory

Roger Gregory, c/o XOC, Box 7615, Ann Arbor, MI 48017.

The Xanadu Hypertext System is a unified system for storing and managing certain forms of information, namely: bodies of documents. It is built around a small number of relatively simple concepts:

- DOCUMENTS
- Multiple VERSIONS of Documents
- Distributed Storage and Distributed Computation – NETWORKING between main frames and microcomputers
- Isolation of storage functions from display functions–BACKEND/FRONTEND

I will discuss each of these in turn and thereby build up a picture of what the Xanadu System is, what it does, and what makes it unique.

Documents

A "document" is a chunk of contiguous characters (or other data) which functions in some way as a conceptual unit. It is simply a piece of text which some user has declared to be a single thing. It corresponds roughly to the notion of a "file" as found in most computer systems. Beware of this analogy, however, because it breaks down quite thoroughly in the face of the other properties of the system.

A document may contain whatever the user decides to put in it. While documents will generally contain text, the system assumes nothing about their contents. Any arbitrary set of data which can be represented as a string of bytes may be stored in a document. Possible documents include letters, books, chapters of books, memos, marginal notes, musical scores, reports, pictures, computer programs, maps, diagrams, recordings, etc.—whatever the user desires the document to be.

Each document stored in the system is assigned a unique identifier when it is created. This identifier can then be used in later transactions with the system to designate that particular document. This document identifier is a special sort of number called a "tumbler." Tumblers are multi-part numbers similar to those used to number sections of technical manuals and to index books in the Dewey Decimal System. For example, 1.0.3.15 and 123.5.6391.0.8.1 are both tumblers. They have the property that they can be expanded at any point, and it is therefore possible to implement a numbering system that never runs out of precision. This means that there is no limit to the number of document identifiers possible and therefore no theoretical limit to the number of possible documents due to a finite address space.

Since a document is a linear array of bytes, we can refer to any particular character in a document by its position in the document. Thus, by combining a document's unique identifier with a character position within that document, we can uniquely address any character stored in the entire system.

By designating a particular (starting) character and a length we can address any contiguous string of characters in the system. Such a string is called a "span." By clever use of tumblers instead of plain integers to represent character positions and span lengths, it is possible to have spans which contain whole documents and cross document boundaries.

Versions

The Xanadu Hypertext System is a real and currently available product which can manage multiple differing versions of a single document. Typically, documents will change with time, as they are corrected or revised. Parallel versions may be made, as when writers weigh alternate drafts of a chapter or when a computer program and its documentation are maintained for several different installations.

Whenever a document is changed, the changes are recorded. The system remembers both the old and the new versions and can retrieve either on command. Thus it is possible to retrieve a document in the state it was in at any point in time since its creation. This can be particularly useful, for example, in software development or creative writing when one may make false starts and wish to back up to a previous state. In addition, the system can indicate the changes which caused two alternate versions to differ. Thus a complete "audit trail" through the history of a document can be constructed. Such a trail is in the form of an historical trace tree which branches whenever parallel versions are maintained.

In the general case, the problem of comparing two pieces of text to determine the differences between them is NP-complete. It may seem curious then that the Xanadu System is able to do this. The reason that the system can determine how documents differ is because *it does not compare them.* Rather, *it* performs the actual changes and "remembers" them, instead of having a whole new document passed to it by some external editor, as most systems would do.

Xanadu understands a simple set of editing primitives by which it can be directed to make changes to a document. Those are:

- INSERT characters at some point in the document.
- DELETE characters from some point in the document.
- MOVE a span of characters from one location in the document to another.
- COPY a span of characters from one location in the document (or from anywhere in the rest of the system, for that matter) to some location in the document.

A moment's thought will reveal that these operations are sufficient to perform then more complex editing operations.

This sort of arrangement is not unique. However, most systems which do this, such as Bell Laboratories SCCS (Source Code Control System) and DEC's CMS (Code Management System) merely keep a log of the edit operations and then perform those operations in chronological order to produce the desired version. While this is sufficient in many cases, it is unwieldy when there are many parallel branching versions and it is costly in terms of computational overhead, becoming more so as more and more changes are made to a document.

In the Xanadu System, these primitives correspond to operations on the proprietary data structures in which the documents are stored. These data structures tell the system which characters are in which versions and in what order. This allows the proper retrieval of any version quickly and efficiently. The historical traceback facility is built in implicitly. To the best of our knowledge, the Xanadu System is the only system in existence able to manage multiple versions of data in such a general and efficient manner.

Links

A "link" is merely an explicit logical connection between one piece of data and another. The Xanadu Hypertext System supports a completely generalized

linking facility. Any body of text in the entire system can be linked to any other body of text in the system. Links may connect text within a document or run between documents.

Any given link connects from one set of text in the system to another. The bodies of text at either end of the link are therefore called the link's "end-sets." The first end-set is called the "from-set" and the second is called the "to-set." By convention links have an implied direction from the from-set to the to-set, not surprisingly, although the relationship is entirely symmetrical and a link may be followed in either direction.

The potential contents of an end-set are totally unrestricted. An end-set may consist of an unlimited number of spans. The characters linked to or from need not be contiguous or even in the same document.

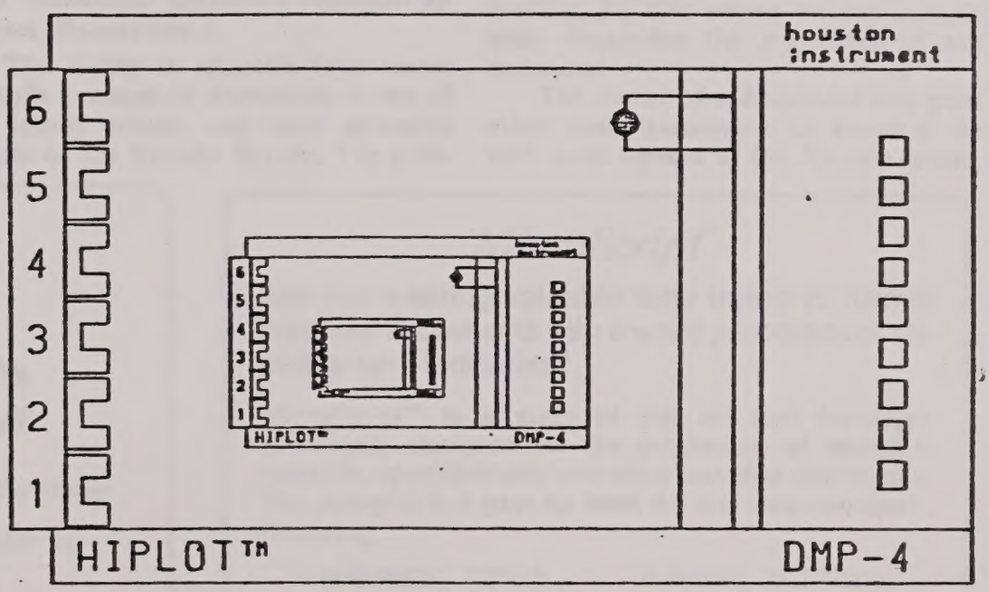
Links themselves are said to reside inside documents, located in a separate logical address space from the text. A link may reside in one document and link from a second document to yet a third. The location of the residence of a link is entirely independent of the contents of the link's end-sets. The fact that links are among the potential contents of documents means that links themselves may be linked from or to, just as characters may. Thus very general sorts of indirect structures may be assembled.

It was mentioned above that the end-sets are symmetric and only distinguished by convention. In fact, a link may have any number of end-sets. The use of these extra end-sets may vary depending on the purpose of the link. We have found it convenient to define, again by convention, a third end-set called the "three-set."

The purpose of the three-set is to designate information pertaining to the link itself. By this means we can define "link types" which may in turn be used to determine how the link is to be interpreted. For example, we might create a "quotation link" type, which designates that the passage linked to is to be displayed in quotation marks at the location linked from. A "footnote link," on the other hand, might simply designate that a superscripted asterisk be shown at the "from" location and that the to-set contents be displayed only when the user so requests. In a more advanced system, the three-set might point to a program or subroutine to be downloaded into the display computer. This program would then decide how to display the link and its end-sets. In any case, the mechanism is totally general and constrained only by convention.

There are very few systems which attempt to implement a link facility at all like this. We are aware of only one system which even comes close, Englebart's "Augment," available on Tymenet. Augment implements links by embedding code sequences directly into the text. This technique has a number of drawbacks: it rapidly becomes unworkable when more than a small number of links are defined, it is very difficult to maintain the bi-directionality of links using such a scheme, discontinuous end-sets are just about impossible, and embedded code sequences would interfere with the

QCAL™ DEVICE-INDEPENDENT CP/M GRAPHICS



QCAL(tm) emulates the Calcomp(tm) Basic Subroutine Package. The QCAL user (with Microsoft(tm) FORTRAN) may employ the industry standard calls (PLOT, AXIS, SYMBOL, etc.) but utilize many available graphic output devices. A generation of prior graphics application software becomes accessible under CP/M(tm), and new programs using QCAL handle graphics in a time-proven, standardized, and transportable manner. Metric capability is built-in.

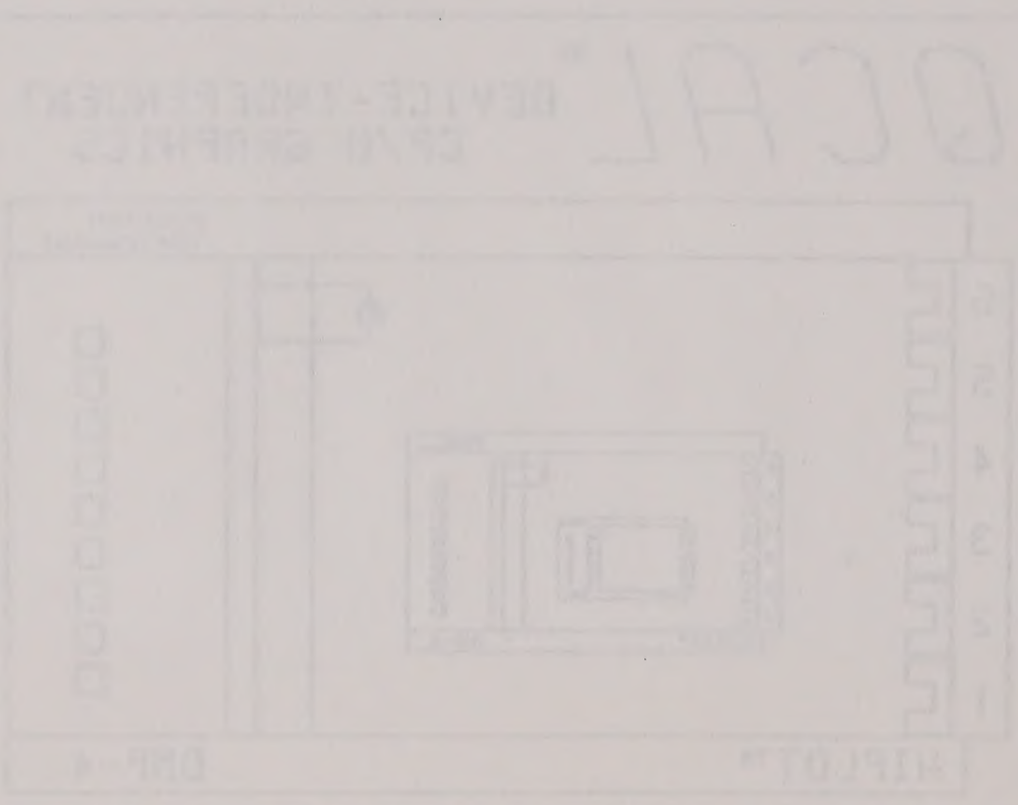
QCAL includes (at $295 on 8" SD diskette) user's manual, sample programs, relative object code for the emulated calls, source for fonts (7 alphabets for USA and Western Europe), and source for one sample graphics device driver. Available choices are HIPLOT(tm), Watanabe(tm), and NEC Spinwriter(tm). Custom fonts and drivers are easily created using supplied documentation.

QCAL was featured in the US exhibit at "Europe Software 82", The Netherlands. It is now in use around the world.

QCAL (tm) Tesseract Associates; Calcomp (tm) California Computer Products, Inc.; Microsoft (tm) Microsoft Corp.; CP/M (tm) Digital Research, Inc.; HIPLOT (tm) Houston Instrument division of Bausch & Lomb; Watanabe (tm) Watanabe Instrument Corp.; Spinwriter (tm) NEC Information Systems, Inc.

TESSERACT ASSOCIATES
STINSON LAKE ROAD
RUMNEY, NH 03266 (USA)
(603)-786-9561. (617)-964-6740

Circle no. 20 on reader service card.



versioning mechanism. As with versions, proprietary data structures enable our system to work. Xanadu links are not embedded in the text. Although links are said to reside in documents, this is simply a means to allow links to be addressed and thus linked to. There are a number of tricky problems that a linkage system such as this presents. In particular, answering the question, "What links connect from other places to a particular location?", is very difficult using most conventional schemes of organizing data. Consider, for example, the problems inherent in assembling such things as citation indices. Nevertheless, we believe we have the general-case solution to these problems.

One might ask what happens to the end-sets of links when the documents containing the characters in the end-sets are changed. There is a very close relationship between the linking facility and the versioning facility. When new versions of a document are created, links are preserved. It is possible to link to a particular version of a document, to all versions, or to the most current version. Links will always point to the same text even though the document containing the text may change. For example:

Here is some text.	Some text in a document.
Here is some more text.	Some characters are inserted.
Here is me more text.	Some characters are deleted.

[The underlined characters represent an end-set of some link.]

The ability to maintain links across multiple versions of documents is one of the totally unique and most powerful aspects of the Xanadu System. The problems inherent in doing this are almost unimaginably subtle, but we believe we have found the solution.

Multi-user

The facilities of the Xanadu Hypertext System would be of limited utility if each user functioned in isolation. Xanadu is therefore designed to be a multi-user system, allowing independent users to access and manipulate the same body of information.

Each user in the system has a unique user identifier tumbler. Each document in the system has an owner, and the owner's user identifier tumbler comprises one of the fields of the document identifier tumbler. By this means we can immediately determine the ownership of any document.

The owner of a document may grant other users permission to access it. As with most aspects of the Xanadu system,

8087
&
IBM® PC

The 8087 Assembler Support System

- Extends your IBM Macro Assembler to include the 8087 Instruction Set.
- Adds 77 new instructions and 6 new data types.
- Supports all 8086 Addressing Modes.
- Includes a 100 page Users Manual.
- Requires no preprocessing, linking or special compiles.
- Uses Intel® Mnemonics.

$49.95 **COMPLETE!**
Arizona Residents
Add 4% Sales Tax

SEND CHECK OR MONEY ORDER TO:
Southwestern Data Processing
Box 40876 Tucson, Arizona 85717
(602) 881-4454

IBM PC IS A REGISTERED TRADEMARK OF IBM
INTEL IS A REGISTERED TRADEMARK OF THE INTEL CORP.

Circle no. 21 on reader service card.

MicroScript™

Are you wasting valuable time trying to format complex documents with a word processor or obsolete text formatter?

MicroScript™ is a state of the art text formatter specifically designed for the production of technical manuals, specifications, and other complex documents. This powerful tool pays for itself the first time you use it. Featuring:

- generalized markup
- left alignment
- center alignment
- right alignment
- justification
- left indention
- right indention
- bold text
- underscored text
- proportional spacing
- fully definable page
- multiple columns
- headers and footers
- floating text blocks
- footnotes
- variable line spacing
- widow supression
- section numbering
- imbedded documents
- automatic lists
- macro processing
- symbol processing
- table of contents
- direct printer control
- initialization profile
- page numbering

$99 postpaid within U.S., outside U.S. add $10. CA residents add 6%. Specify CP/M-80*, CP/M-86*, MS-DOS*, or PC-DOS*; printer type; disk format.

Software Technique™
6531 Crown Blvd., Suite 3A
San Jose, CA 95120
(408) 997-5026

* CP/M-80, CP/M-86 trademarks of Digital Research, MS-DOS trademark of Microsoft, PC-DOS trademark of IBM Corporation.

Circle no. 22 on reader service card.

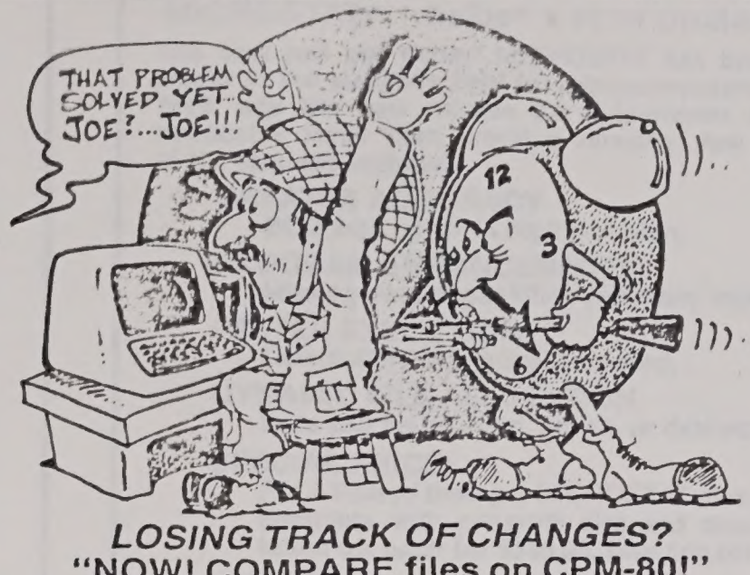
the access privilege mechanism is as general as possible. Permission to access a particular document may be granted to an arbitrary number of individuals, to groups of users,or to the entire user community. Type of access permitted (Read, Write, etc.) is not a relevant concept in the Xanadu System. You may always edit another user's document to which you have access, but in doing so you create a new version which is your own document. By this means we can immediately determine the ownership of any document.

The multi-user capability of Xanadu makes a number of useful activities possible. For example, you can write marginal notes commenting on someone else's work. You write your comments in a document of your own and then link them to the other person's document. The other user need never be aware of this. It is like scribbling in a library book without defacing the book to the eyes of others. As another example, it is possible, using documents, links, and the permission facility, to create a quite sophisticated "electronic mail" system. By passing a link to another user you can point him at something you want him to read.

Networking

As the number of users on an installation of the Xanadu System grows larger, the processing and storage capability of even the largest and fastest computer will eventually be used up. The only viable solution to the fundamental resource limitations of centralized facilities is distributed processing and storage. The Xanadu System is designed to be able to grow arbitrarily large without having a significant adverse impact on overall system performance. It does this by allowing the system to spread out over as many machines as are needed to support all users effectively.

The portion of the system which stores and retrieves documents and links is designed to be able to run simultaneously on an arbitrary number of machines which are tied together in a network by some sort of communication medium. If a piece of information is requested at one node of the network but is not stored at that node, the Xanadu software running on that node's computer sends a request via the network to the machine in which the information is stored. This machine in turn sends back the requested data.



LOSING TRACK OF CHANGES?
"NOW! COMPARE files on CPM-80!"

- Compare Program Versions
- Fast, NO File Restrictions
- Side by Side or Vertical Listings of Differences
- Output to File or Printer
- Compare Documents
- Extensive User Options: Ignore Comments, Blank lines and more
- Create Documents with Change Bars
- Sensible Command Line defaults

COMPARE is a superb software tool with excellent features for the serious professional programmer with no time to waste. Document mode with Change Bars designed especially with Writers in mind.

Call or Write for Information ..... Only $95

SOLUTION TECHNOLOGY, INC.

Suite 218 • 1499 Palmetto Park Rd. • Boca Raton, FL 33432 • (305) 368-6228

*CPM-80 is a trademark of Digital Research Inc. *Check or COD. Florida residents add 5% sales tax.

Circle no. 23 on reader service card.

MCDISPLAY™

$175.00

THE BEST MBASIC DISPLAY INTERFACE EVER DEVELOPED!

Let MCDISPLAY handle the interface to the program user in your application program. For CP/M.

ORDER YOUR COPY TODAY

CALL COLLECT (803) 244-8174

DEMO PACKAGE $10.00 MANUAL $25.00

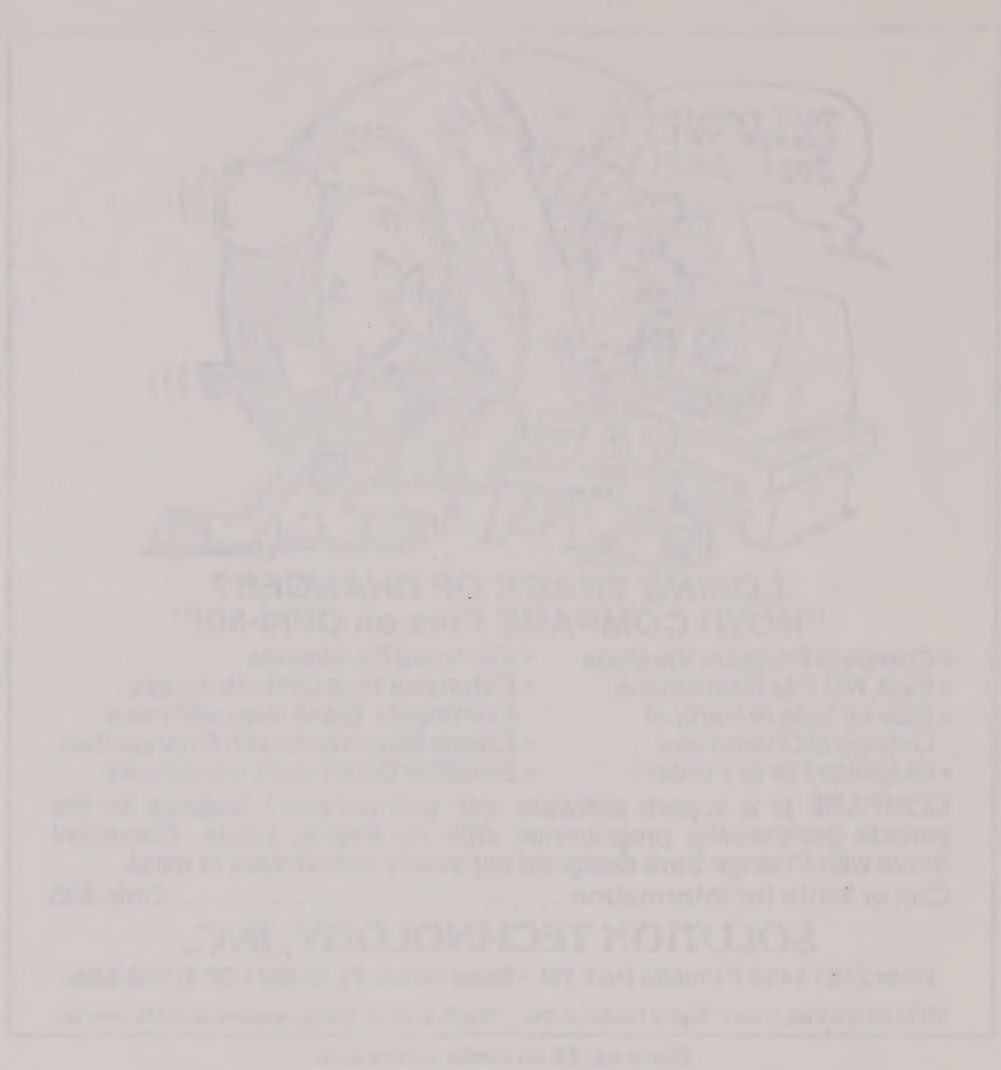
CHECK, MONEY ORDER, P.O., VISA, MASTERCARD

MasterComputing Inc.

P.O. Box 17442
Greenville, SC 29606
(803) 244-8174

CP/M is a trademark of DIGITAL RESEARCH
MBASIC is a product of Microsoft

Circle no. 24 on reader service card.



Professionals Prefer Q/C.

For only $95, Q/C is a professional, fully-supported C compiler for CP/M. Q/C supports a large subset of C, and is upward compatible with the UNIX Version 7 C compiler from Bell Labs. The Q/C library includes over 50 input/output and other support functions, all written in C.

When you buy Q/C, you get a working compiler that generates assembly language. You also receive the complete source code for the Q/C compiler and the function library. The Q/C compiler is written in C, with a few functions hand-coded in assembler to enhance performance. Most compiler options can be customized to suit your taste by using the configuration program we supply.

What really sets Q/C off from the competition is our 138-page *User's Manual*. The tone of the manual is informal and personal. Jim Colvin (the author of Q/C) tells you how to use the compiler, and clearly describes each library function. There's even a chapter that explains in detail the "internals" of Q/C.

Q/C is a fully-supported professional product. We continue to develop and enhance Q/C, and provide updates at a nominal cost. Write or call for details of Q/C Version 2.0.

THE CODE WORKS
5266 Hollister
Suite 224
Santa Barbara, CA 93111
(805) 683-1585

CP/M is a trademark of Digital Research
UNIX is a trademark of Bell Laboratories.

Circle no. 50 on reader service card.

MICROSTAT® - Release 3.0

MICROSTAT® + baZic® = PERFORMANCE

The best just got better! MICROSTAT has been the leader in the statistics field for microcomputers since 1979, and the new release 3.0 outperforms and is noticeably faster than previous versions. Just a few of the features include:

GREATER ACCURACY
BCD with up to 14 digit precision;

PROGRAM ENHANCEMENTS
Missing data capabilities and many more;

FASTER EXECUTION
Calculation time greatly reduced;

DYNAMIC FILE ALLOCATION
Data can be inserted, added, or deleted;

SPECIAL PRICE:
For a limited time get MICROSTAT plus baZic complete with program disk and documentation for each for $395.00, save $50.00!

The MICROSTAT - baZic version requires: a Z80 CPU, CP/M™ and 48K of memory. Available formats: 8" SD disk or 5¼" North Star only. Check with your dealer for other formats. Also available for: Microsoft's Basic-80™, North Star DOS and IBM. For more information, call or write:

ECOSOFT INC.
P.O. Box 68602
Indianapolis, IN 46268-0602
(317) 255-6476

MICROSTAT is a registered trademark of ECOSOFT, INC.
baZic is a registered trademark of MICROMIKES, INC
CP/M is a registered trademark of DIGITAL RESEARCH
Basic-80 is a registered trademark of MICROSOFT

Circle no. 49 on reader service card.

The system does not care what communications technology is used, as long as it gets the bits from one place to the other. No particular network topology is assumed, as long as paths exist between any nodes which might wish to communicate. Nodes may be added to or deleted from the network continuously with the minimum possible disruption. If a node is deleted, data that was stored solely on that node becomes unavailable, of course, and any communication paths that went through that node may need to be rerouted, possibly delaying some transmissions.

Communications routing from one node to another is automatic and implicit in the network design. Nodes are not required to have full knowledge of the entire network, thus the network can grow without every node being informed of every change to the network topology.

Various optimizations are built in. If a node finds that it is being given frequent requests for data stored on a distant node, it can make a copy of that data and store it locally. The system does not mind that there may be multiple copies of some information. If a request for such information is made, the system tries to get it from the closest or most quickly accessible source. If changes are made to one copy, the system keeps track. All of this is accomplished internally by the same data structures that manage all the other functions of the system.

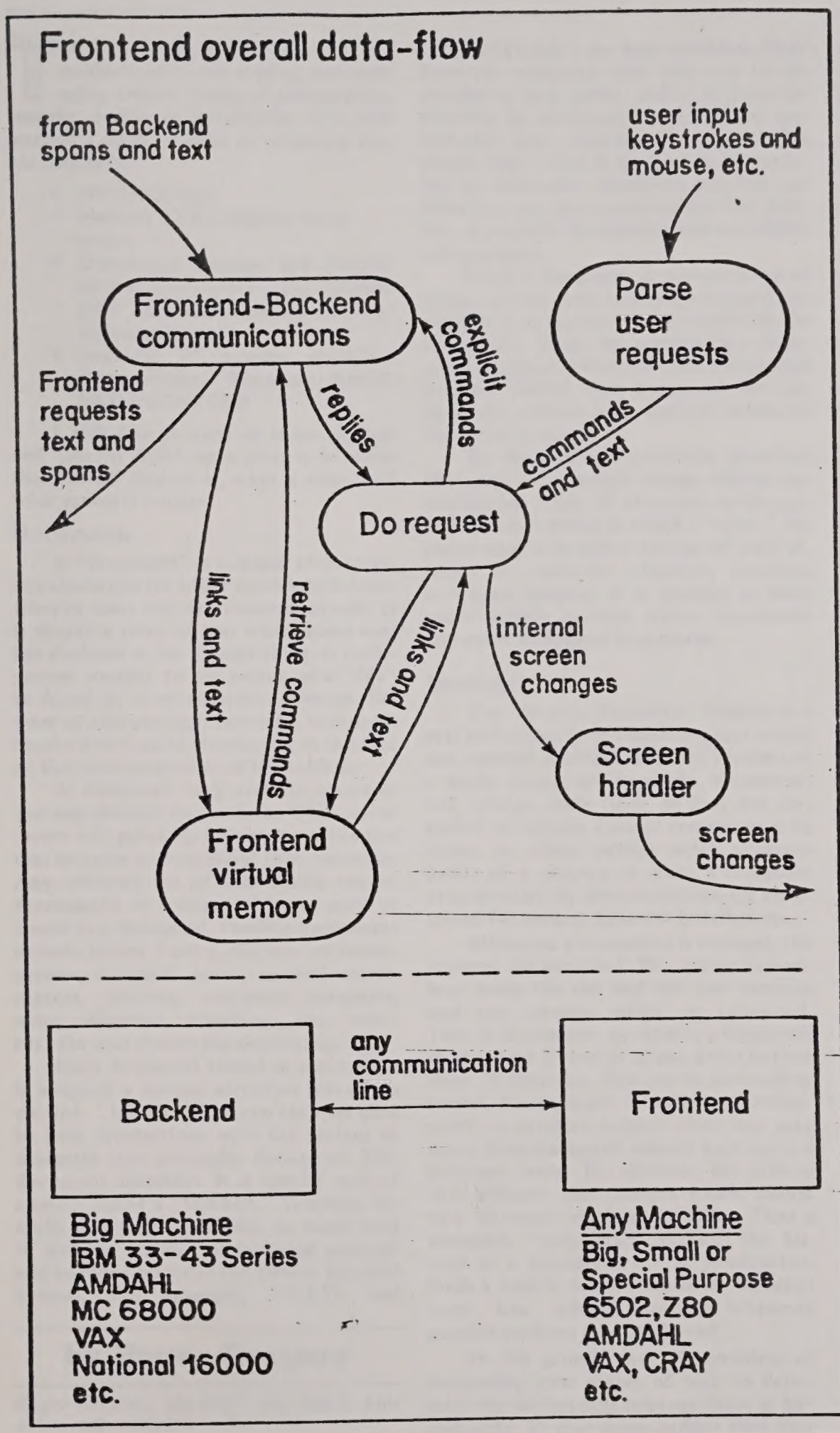
Backend/Frontend

All of the capabilities described above are part of what is called the "backend." The backend handles storage, retrieval, versioning, linking and editing of data. However, it "knows" nothing about the data, *per se*. The backend is the portion which does all the "impossible" things. It contains the proprietary components which we have developed.

User interaction—the display of old data and the acceptance of new data—is accomplished by the other major portion of the system, the "frontend." The frontend worries about whether the data is text or pictures or music or whatever. The frontend decides how the data is to be formatted for display. The frontend knows how to interface with various input or output devices.

The frontend and the backend communicate using a standardized request and data transmission protocol. The frontend sends commands and data to the backend which responds with other data and status information. The specifications for this protocol are free for the asking to anyone. The frontend will, in general, be on a separate computer which is connected to the backend by a communications line (not to be confused with the communications lines which connect the Xanadu network—the whole network *is* the backend). Any program which understands the protocol, running on any computer connected to the backend, can be a frontend. The frontend need not be designed by us, written by us, or purchased from us. While we will be producing and selling frontends, we expect others to do so as well.

Frontends may be general purpose or special purpose. Different applications, different forms of data, different display and input devices and different user preferences can all be accommodated. For example, one might create an office automation frontend which handles text processing, electronic mail and project management functions through Xanadu, and does accounting and database stuff locally. A software development frontend might use Xanadu to manage source and object code and documentation and have local capabilities for compiling, executing and debugging programs. As a third example, a CAD/CAM frontend might use Xanadu to store drawings and documentation and have simulation and modeling functions built in. All of these different activities can be accommodated using the same backend design—the backend is application independent.



Summary

The Xanadu Hypertext System is one of the most powerful systems in existence for managing text in a micro-to-mainframe database environment. It can store arbitrary numbers of documents and retrieve them on demand. It can organize them by using a highly generalized link facility and by allowing (and keeping track of) multiple users on the same system, and allow those users to work with each other on a common document base. It can grow indefinitely over a large distributed network with minimal degradation in performance. It can accommodate almost any conceivable display or input device or mode of user interaction by separating these functions into an independent frontend.

DDJ

Reader Ballot
Vote for your favorite feature/article.
Circle Reader Service No. 304

CBASIC* USERS

Now it is possible to recover a ".BAS" from an ".INT" file. Send me a SSSD 8" CP/M * disk with a ".INT" file on it: I will return it with the reconstructed ".BAS" file added. (Multiple ".INT" files on a disk are allowed, not to exceed 48K per disk.)

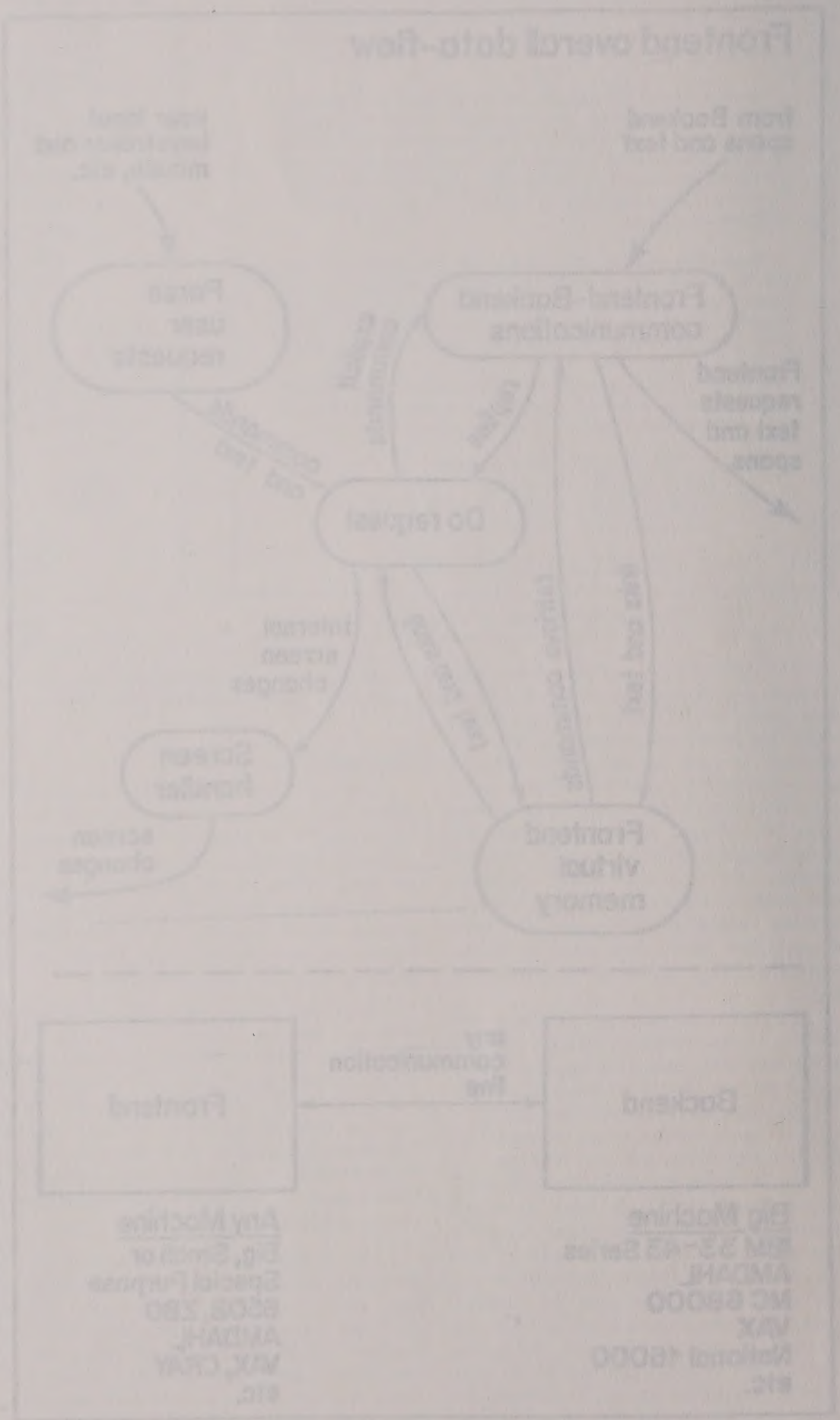
Cost is $40. per ".INT" file. Discount of 10% for 5 to 9, 15% for 10 or more, ".INT" files shipped as a single order.

MC/VISA HONORED ● N.J. RESIDENTS ADD 5%

PETER INGERMAN
40 NEEDLEPOINT LANE
WILLINGBORO, N.J. 08046

**CBASIC and CP/M are trademarks of Digital Research, Inc.*

Circle no. 27 on reader service card.



DUO-TANG
50125
MADE IN U.S.A